1. A branch target address cache (BTAC) for providing a speculative target address to address selection logic, the address selection logic selecting a fetch address for addressing a line in an instruction cache, the BTAC providing the speculative target address based on a presumption that a branch instruction is present in the cache line, the BTAC comprising:

an array of storage elements, configured to cache target addresses of previously executed branch instructions;

an input, coupled to said array, for receiving the fetch address, to index into said array of storage elements to select one of said target addresses; and

an output, coupled to said array, for providing said one of said target addresses indexed by the fetch address to the address selection logic;

wherein said output provides said one of said target addresses to the address selection logic for selection as a subsequent fetch address whether or not a branch instruction is present in the

line of the instruction cache addressed by the fetch address.

2. The branch target address cache of claim 1, wherein said array of storage elements is further configured to store speculative branch information associated with said previously executed branch instructions.

3. The branch target address cache of claim 2, further comprising:

a second output, coupled to said array, for providing a portion of said speculative branch information to control logic for controlling the address selection logic in response to said portion of said speculative branch information.

4. The branch target address cache of claim 2, wherein said speculative branch information comprises information predicting whether the branch instruction presumed present in the cache line will be taken.

5. The branch target address cache of claim 4, wherein said information predicting whether the presumed branch instruction will be taken comprises a taken/not taken bit.

6. The branch target address cache of claim 4, wherein said information predicting whether the presumed branch instruction will be taken comprises a plurality of bits.

7. The branch target address cache of claim 6, wherein said plurality of bits is stored in a saturating up/down counter.

8. The branch target address cache of claim 3, wherein said portion of said speculative branch information comprises an indication of whether said one of said target addresses is a valid target address.

9. The branch target address cache of claim 8, wherein said indication is populated to indicate said one of said target addresses is a valid target address in response to execution of the presumed branch instruction, wherein said one of said target addresses is resolved.

10. The branch target address cache of claim 8, wherein said indication is populated to indicate said one of said target addresses is not a valid target address in response to detecting said one of said target

addresses is erroneous subsequent to said providing said one of said target addresses on said output.

11. The branch target address cache of claim 2, wherein said speculative branch information comprises information specifying a location within the cache line of the branch instruction presumed present in the cache line.

12. The branch target address cache of claim 2, wherein said speculative branch information comprises a length of the branch instruction presumed present in the cache line.

13. The branch target address cache of claim 2, wherein said speculative branch information comprises an indication of a type of the branch instruction presumed present in the cache line.

14. The branch target address cache of claim 13, wherein said indication of said type of the branch instruction indicates whether the branch instruction is a call instruction.

15. The branch target address cache of claim 13, wherein said indication of said type of the branch instruction

indicates whether the branch instruction is a return instruction.

16. The branch target address cache of claim 2, wherein said speculative branch information comprises an indication of whether the branch instruction presumed present in the cache line spans more than one line in the instruction cache.

17. The branch target address cache of claim 1, wherein each of said storage elements is configured to cache a plurality of target addresses.

18. The branch target address cache of claim 1, wherein the branch target address cache is external to the instruction cache.

19. A branch target address cache (BTAC) for caching characteristics of branch instructions only, the characteristics including a branch target address and prediction information, the BTAC comprising:

an input, for receiving a fetch address that accesses an instruction cache that is external to the BTAC;

an array of storage elements, coupled to said input and indexed by said fetch address, for caching characteristics of branch instructions only; and

an output, coupled to said array, for providing a branch target address when said input receives said fetch address;

wherein said branch target address is provided to said instruction cache as a subsequent fetch address.

20. A pipelined microprocessor having a branch target address cache, the microprocessor comprising:

first cache lines, within the branch target address cache, for caching branch target addresses;

second cache lines, within an instruction cache, for caching instructions;

wherein said first cache lines and said second cache lines are coupled to a fetch address bus that provides a fetch address for indexing into both of said first and second cache lines; and

wherein the number of said first cache lines is less than the number of said second cache lines.

21. A pipelined microprocessor having separate caches for instructions and branch target addresses, the microprocessor comprising:

a first plurality of cache lines for storing instruction bytes, said first plurality addressed by a fetch address on a fetch address bus; and

a second plurality of cache lines, coupled to said fetch address bus, for storing branch target addresses that are addressed by said fetch address.

22. The microprocessor of claim 21, wherein said first and second plurality of cache lines are physically distinct.

23. The microprocessor of claim 21, wherein said fetch address is a virtual address.

24. The microprocessor of claim 23, wherein said first plurality of cache lines are comprised in an instruction cache that has logic for translating said virtual fetch address to a physical fetch address, wherein said second plurality of cache lines are comprised in a branch target address cache (BTAC) that

does not include logic for translating said virtual fetch address to a physical fetch address.

25. The microprocessor of claim 24, wherein said instruction cache provides one of said first plurality of cache lines of instruction bytes selected based on said physical fetch address, wherein said BTAC provides one of said target addresses based on said virtual fetch address.

26. The microprocessor of claim 21, wherein the microprocessor speculatively branches to one of said target addresses addressed by said fetch address even though one of said first plurality of cache lines of instruction bytes addressed by said fetch address has been modified since said one of said target addresses was cached in said second plurality of cache lines such that no branch instructions are present in said addressed one of said first plurality of cache lines.

27. The microprocessor of claim 21, wherein the microprocessor is configured to speculatively branch to one of said branch target address addressed by said fetch address in response to a hit of said fetch address within said second plurality of cache lines whether or not a branch instruction is cached within

one of said first plurality of cache lines of instruction bytes selected by said fetch address.

28. The microprocessor of claim 21, wherein a possibility exists of virtual aliasing of said fetch address between said first and second plurality of cache lines.

29. The microprocessor of claim 21, wherein said first plurality of cache lines provides an instruction in response to said fetch address, wherein the microprocessor speculatively branches to one of said branch target addresses selected by said fetch address erroneously because said instruction is not a branch instruction.

30. The microprocessor of claim 21, further comprising:

an instruction buffer, coupled to said first plurality of cache lines, for buffering said instruction bytes received from said first plurality of cache lines, wherein said instruction buffer operates in conjunction with said second plurality of cache lines to accomplish substantially zero penalty speculative branches.

31. A pipelined microprocessor comprising:

an instruction cache that is indexed by a fetch address, said instruction cache for caching instructions, and for providing said instructions to an instruction buffer;

a branch target address cache, coupled to said instruction buffer and indexed by said fetch address, for caching branch target addresses;

said instruction buffer comprising a plurality of hit indicators that are associated with said instructions, said indicators specifying whether the microprocessor has speculatively branched to one of said branch target addresses.

32. The microprocessor of claim 31, wherein said instruction buffer includes one of said plurality of hit indicators associated with each byte of said each of said instructions stored in said instruction buffer.

33. The microprocessor of claim 31, wherein said instruction cache and said branch target address cache are accessed substantially in parallel.

34. A method of speculatively branching in a pipelined microprocessor, comprising:

caching a plurality of branch target addresses in a branch target address cache (BTAC);

accessing said BTAC with a fetch address of an instruction cache after said caching;

determining whether said fetch address hits in said BTAC in response to said accessing; and

branching the microprocessor to one of said plurality of branch target addresses selected by said fetch address if said fetch address hits in said BTAC whether or not a branch instruction is cached in a line of said instruction cache indexed by said fetch address.

35. The method of claim 34, further comprising:

storing a branch direction prediction associated with each of said plurality of branch target addresses prior to said accessing said BTAC.

36. The method of claim 35, wherein said branching the microprocessor to said one of said plurality of branch target addresses selected by said fetch address is

performed only if said associated branch direction prediction indicates said branch instruction will be taken.

37. The method of claim 34, further comprising:

storing an indication said branching was performed if said branching is performed.

38. The method of claim 37, wherein said storing said indication said branching was performed comprises storing said indication in an instruction buffer.

39. A method for speculatively branching in a pipelined microprocessor, comprising:

providing a cached speculative branch target address without having decoded an instruction for which said speculative branch target address is cached;

providing a stored speculative branch direction without having decoded said instruction for which said speculative branch direction is stored;

speculatively branching the microprocessor to said speculative branch target address if said speculative branch direction indicates said instruction will be taken.

40. A branch target address cache for speculatively predicting target addresses of branch instructions cached in an instruction cache, comprising:

an input, for receiving a fetch address of the instruction cache;

an array of storage elements coupled to said input, each configured to cache a target address of a branch instruction; and

an output, coupled to said array, for providing said target address cached in one of said array of storage elements indexed by said fetch address;

wherein said output provides said target address without said branch instruction having been decoded by a microprocessor comprising said branch target address cache.

41. A pipelined microprocessor for speculatively branching, comprising:

an instruction cache, indexed by a fetch address provided on a fetch address bus, said instruction cache providing a line of instructions to instruction decode logic;

said instruction decode logic configured to decode said line of instructions subsequent to said instruction cache providing said line of instructions; and

a branch target address cache, coupled to said fetch address bus, configured to receive said fetch address and in response thereto to provide a speculative target address as a subsequent fetch address on said fetch address bus;

wherein the microprocessor is configured to speculatively branch to said speculative target address prior to said instruction decode logic decoding said instruction.

42. The microprocessor of claim 41, wherein said instruction decode logic decodes said line of instructions subsequent to the microprocessor

speculatively branching to said speculative target address and said instruction decode logic determines that no branch instructions are present in said line of instructions.